ISSN:0975-9646

A new horizon in data security by Cryptography & Steganography

(1) Dipti Kapoor Sarmah
Computer Engineering Dept.
Maharashtra Academy of Engineering, Pune.
dkapoor@maepune.com

(2) Neha Bajpai
Computer Engineering Dept.
Center for Development of Advance Computing, Noida
nehakapoor@cdacnoida.in

Abstract

Steganography and Cryptography are two popular ways of sending vital information in a secret way. One hides the existence of the message and the other distorts the message itself. There are many cryptographic techniques available and among them AES is one of the most powerful techniques. In Steganography, various techniques in different domains like spatial domain, frequency domain etc are available to hide messages. It is very difficult to detect hidden message in frequency domain and for this domain various transformations like DCT, FFT, Wavelets etc are available. In this paper we propose ***a new technique*** *in which Cryptography and Steganography are used as integrated part along with newly enhanced security module for transmitting text. Before encryption we are using Huffman compression technique to compress the input plain text and compressed text is encrypted using advanced encryption standard (AES. After encryption, a part of the compressed encrypted message (cipher) is hidden in DCT of an image; remaining part of the cipher is used as a secret key to regenerate the cipher text. Introduction of compression technique improves the data hiding capacity and partial cipher hiding in stego-image make this system more secured.*

Keyword: *Cryptography, Steganography, Stego- image, Compression, Threshold Value, DCT Coefficient*

1. Introduction

Cryptography [1] is the widely used well known technique that manipulates information (messages) in order to cipher and Steganography [1] is the art and science of communicating in a way which hides the existence of the communicated message. Cryptography scrambles a message so it cannot be understood and the Steganography hides the message so it cannot be seen. In this paper one new system is proposed, which uses **Compression[2] with Cryptography and Steganography** for better confidentiality, security and achieves a faster encryption and decryption method. With the help of this system we can hide more data in an image and decrease the transmission time as well .Presently we have very secure methods for both cryptography and Steganography – AES algorithm is a very secure technique for cryptography and the techniques which use frequency domain are considered highly secured for

steganogarphy. But even if we combine these techniques straight forwardly, there is a chance that the intruder may detect the original message after couple of attacks. **Therefore, our idea is to propose a new system which use the Huffman compression technique[3] for compression of the plain text and the compressed text is encrypted using AES algorithm. The output of the encrypted text which is in the form of hexadecimal values will be hidden in an image through steganography (DCT technique) with more security levels to get a very highly secured system for data hiding.** This paper mainly focuses on to develop a new system with extra security features where a meaningful piece of text message can be hidden by combining three techniques like Compression, Cryptography and Steganography.

As we know -

- Hiding data is better than moving it shown after encryption.
- To hide data in a popular object that will not attract any attention.
- In case the data is extracted, it will be in the compressed form.
- After decompression only we can have the encrypted data.
- With the help of our new system we can have more data in an encrypted form.

If we hide the encrypted message directly in some medium still there is a chance that the intruder can break the code. In our new system instead of applying existing techniques directly we will be using the following approach –

- Instead of hiding the complete compressed encrypted text into an image, we will be hiding a part of the compressed encrypted message.
- Unhidden part of the compressed encrypted message will be converted into one secret key.
- In this system to get the original message one should know, along with keys for Compression, Cryptography and Steganography, one extra key and the reverse process of the key generation.

So our final goal of the paper is to propose a new system which is highly secured and even if somebody retrieves the message from stego image [4] it becomes a meaningless for any existing cryptographic techniques.

2. Basic Concepts and Related work

There are many aspects to security and many applications. One essential aspect for secure communications is that of cryptography. But it is important to note that while cryptography is necessary for secure communications, it is not by itself sufficient. When we encrypt the data using cryptography, any length of data can be encrypted. But it is very difficult to hide the whole encrypted data in an Image using DCT technique in Steganography so we use compression technique first, before encrypting the data.

There are many compression techniques:

a) **Huffman Compression**
b) **LZW compression**
c) **Arithmetic coding and so on**.

In our paper we are using Huffman compression method.

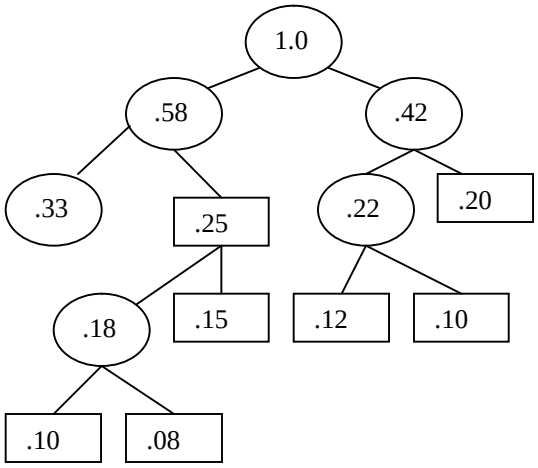
2.1 Huffman Compression[5]

It is a method for the Construction of Minimum-Redundancy Codes. It uses a specific method for choosing the representation for each symbol, resulting in a prefix code (sometimes called "prefix-free codes", that is, the bit string representing

some particular symbol is never a prefix of the bit string representing any other symbol) that expresses the most common source symbols using shorter strings of bits than are used for less common source symbols.
Huffman compression belongs into a family of algorithms with a variable codeword length. That means that individual symbols (characters in a text file for instance) are replaced by bit sequences that have a distinct length. So symbols that occur a lot in a file are given a short sequence while other that are used seldom get a longer bit sequence. The basic idea behind the algorithm is to build the tree bottom-up whose leaves are labeled with the weights. When the Huffman algorithm is used to construct a code, the weights represent the probabilities associated with the source letters. Initially there is a set of singleton trees, one for each weight in the list. At each step in the algorithm the trees corresponding to the two smallest weights, $w(i)$ and $w(j)$, are merged into a new tree whose weight is $w(i)+w(j)$ and whose root has two children which are the sub trees represented by $w(i)$ and $w(j)$. The weights $w(i)$ and $w(j)$ are removed from the list and $w(i)+w(j)$ is inserted into the list. This process continues until the weight list contains a single value. If, at any time, there is more than one way to choose a smallest pair of weights, any such pair may be chosen. In the Huffman algorithm, the process should begin with a non increasing list of weights because it provides a more efficient implementation. One example is shown:

A1	0.25	0.25	0.25	0.33	0.42	0.58	1.0
A2	0.20	0.20	0.22	0.25	0.33	0.42	
A3	0.15	0.18	0.20	0.22	0.25		
A4	0.12	0.15	0.18	0.20			
A5	0.10	0.12	0.15				
A6	0.10	0.10					
A7	0.08						

(a)



(b)

2.1.1 Advantages of using Compression

1. Algorithm is easy to implement
2. Produce a lossless compression.
3. If we use compression before AES then hidden plain text volume will be more in an image indirectly.

2.2 Cryptography Techniques

After applying the Huffman compression algorithm we need to know about the Cryptographic algorithms. There are some specific security requirements [6] for cryptography, including Authentication, Privacy/confidentiality, and Integrity Non-repudiation. The three types of algorithms are described:

(i)Secret Key Cryptography (SKC): Uses a single key for both encryption and decryption
(ii)Public Key Cryptography (PKC): Uses one key for encryption and another for decryption
(iii)Hash Functions: Uses a mathematical transformation to irreversibly "encrypt" information.

2.2.1AES algorithm for Cryptography

This standard specifies the Rijndael algorithm [7], a symmetric block cipher that can process data blocks of 128 bits, using cipher keys with lengths of 128, 192, and 256 bits. The input, the output and the cipher key for Rijndael are each bit sequences containing 128, 192 or 256 bits with the constraint that the input and output sequences have the same length. In general the length of the input and output sequences can be any of the three allowed values but for the Advanced Encryption Standard (AES) the only length allowed is 128.

2.2.2 Advantages of using AES algorithm

1. Very Secure.
2. Reasonable Cost.
3. Main Characteristics:
   I. Flexibility, II. Simplicity

2.3 Steganography Techniques

Steganography is the other technique for secured communication. It encompasses methods of transmitting secret messages through innocuous cover carriers in such a manner that the very existence of the embedded messages is undetectable. Information can be hidden in images [8], audio, video, text, or some other digitally representative code. Steganography systems[9] can be grouped by the type of covers [10] used (graphics, sound, text, executables) or by the techniques used to modify the covers

a) Substitution system [11].
b) Transform domain techniques [12]
c) Spread spectrum techniques [13]
d) Statistical method[13]
e) Distortion techniques[14]
f) Cover generation methods [14]

We have used DCT frequency domain algorithm in our proposed system.

2.3 DCT [15]-frequency domain algorithm for Steganography

According to the method presented in this paper, the compressed encrypted message is inserted into the DCT domain of the host image. The hidden message is a stream of "1" and "0" giving a total number of 56 bits. The transform is applied to the image as a multiple factor of 8x8 blocks. The next step of the technique after the DCT is to select the 56 larger positive coefficients, in the low-mid frequency range. The high frequency coefficients represent the image details and are vulnerable to most common image manipulation like filtering [16] compression [17] etc. Of course one might argue that this is the place where changes that come from watermarking [18] are more imperceptible, but this is true only if we're speaking of small sized blocks. Our scheme is applied to the whole image and since robustness is the main issue, the low and mid frequency coefficients are the most appropriate. The selected coefficients *ci* are ordered by magnitude and then modified by the corresponding bit in the message stream. If the *i*th message bit ***s(i)*** to be embedded is "1", a quantity *D* is added to the coefficient. This *D* quantity represents the persistence factor. If the message bit is "0", the same quantity is subtracted from the coefficient. Thus the replaced DCT coefficients are

DCT (new) = DCT+1*D
for s(i)=1;
Else
DCT (new) =DCT-1*D
for s(i)=0.

DCT can separate the Image into High, Middle and Low Frequency components. To hide information we need to set a threshold

value [17] for the DCT coefficients depending on the quality of the images.

2.3.2 Advantages of using frequency domain Steganography

- Very secure, hard to detect
- Flexible, different techniques for manipulation of DCT coefficients values

4. A propose technique for combination

The design for the combining two different techniques is purely based on the idea – distort the message and hide the existence of the distorted message and for getting back the original message – retrieve the distorted message and regain the actual message by reversal of the distortion process.
Here we design the system with four modules-

- For Compression – Compression Module
- For Cryptography - Crypto Module
- For Steganography - Stego Module
- For extra security - Security Module

The extra security module that we are providing make this system highly secured. The process flow for the system is as follows-

2.3 Hiding the Text

- **Compressed Module :**

For Compressed Module basically we compressed the Input plain text using Huffman algorithm(Refer **Figure1**).

Figure1: Compressed Module

- **Crypto Module** :

For Crypto Module the following steps are considered for encrypting the data (Refer **Figure2):**

- Insert compressed text for encryption.
- Apply AES algorithm using 128 bit key (Key 1).
- Generate Cipher Text in hexadecimal form.

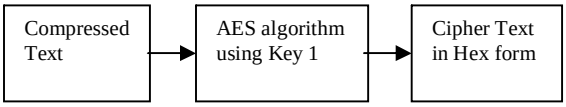


Figure2: Crypto Module

- **Security Module:**

This is an intermediate module which provides an extra security features to our newly developed system. This module is used to modify the cipher text and to generate one extra key. In the reverse process it regenerates the original cipher text (Refer **Figure3).** Before the hiding process this module works as follows:

- Take the first seven hexadecimal values from the cipher. This number of hexadecimal is fixed for the module after performing a series of experiment for better image quality after data hiding in an stego image.
- Remaining hexadecimal values will form a key (Key 3).

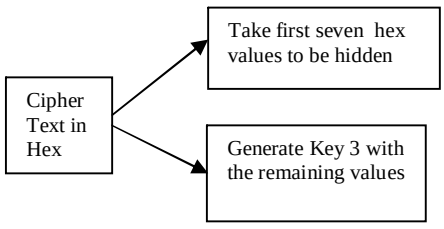


Figure3: Security Module

- **Stego Module:**

For Stego Module the following steps are considered for hiding the above generated Cipher text .For more details refer **Figure4**.

- Take seven hexadecimal values from the above discussed Security Module.
- Scramble these values using a 64 bit key (Key 2).
- Take a Gray Scale Image.
- Find the DCT of the Image.
- Hide the Cipher by altering DCTs.
- Apply Inverse DCT.
- Find the Stego Image.

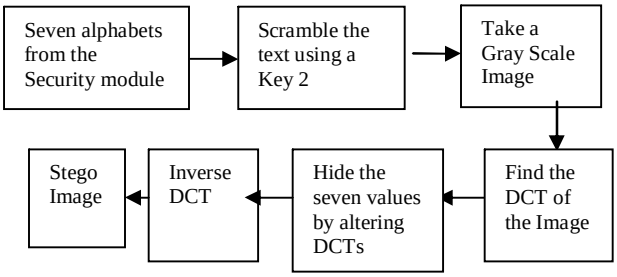


Figure4: Stego Module

2.4 Retrieving Text

- **Stego Module(Reverse Process) :**

For Stego Module the following steps are considered for retrieving the cipher text (Refer **Figure5**):

- Take DCT of the Original Image.
- Take DCT of the Stego Image.
- Take difference of DCT coefficients.
- Retrieve bits of the hidden seven alphabets from LSB of the DCT.
- Construct the distorted seven hexadecimal values.
- Unscrambled the distorted seven hexadecimal values using Key 2.
- Retrieve the original seven alphabets.

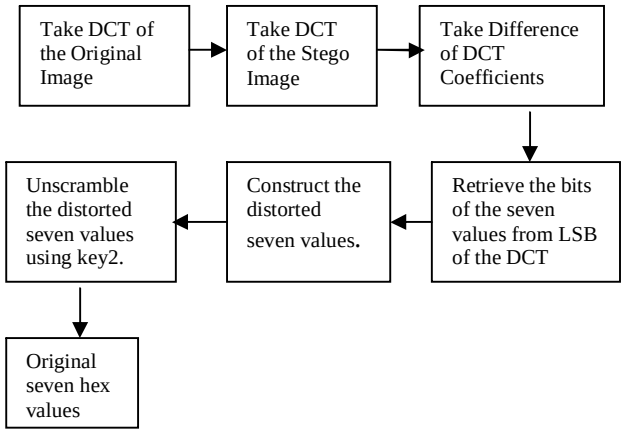
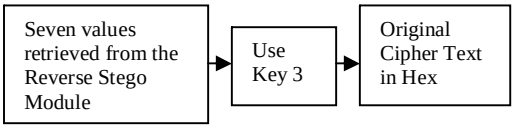


Figure5: Stego Module(Reverse Process)

- **Security Module(Reverse Process):**

For Security Module the following steps are considered for retrieving the cipher text (refer **Figure6):**

- Club the seven hexadecimal values with the rest of the text by using Key 3.



Figur6: Security Module (Reverse Process)

- **Crypto Module(Reverse Process):**

For Crypto Module the following steps are considered for retrieving the original text. For more details refer **Figure7:**

- Get the above retrieved cipher text.
- Reverse AES algorithm by using Key 1.
- Get the Compressed message.

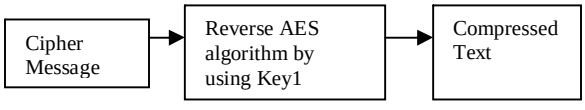


Figure7: Crypto Module (Reverse Process)

• **Decompressed Module**(Refer **Figure8**):

➢ Apply Huffman decompression technique to the text retrieved from the Crypto Module.
➢ Get the above retrieved text.
➢ Apply decompression using Huffman Decompression technique.
➢ Retrieve the original Plain text.

Figure8: Decompressed Module

3. Implementation Details

This new technique is mainly developed in VC6.0 plate-form using VC++. Here mainly four modules involved –

a) Compression Module – Huffman Compression Module
b) Crypto Module - AES Implementation Module
c) Security Module – Newly developed technique for Key generation
d) Stego Module - DCT Techniques Implementation Module

These modules are designed and coded as reusable components and can work independently.

3.1 Tools and Libraries used

i. Arisimage Routines [20]
ii. Cximage599c [21]
These libraries are available with free license.

3.3 Algorithm for the proposed system

Firstly, we compress the Input plain text using Huffman Compression. The compressed form of the text will work as input to the AES algorithm for encryption. In AES Implementation, we get the cipher text in the hexadecimal form and the length of the cipher text is large. So we partially hide the encrypted information in the image and with the help of the remaining part of the encrypted message we generate one key. This key is a secret key and the receiver needs to know this key to retrieve the original encrypted message.
The steps for the algorithm are discussed below (Refer **Figure9 & Figure10**):

3.3.1 Hiding Text

➢ Apply Huffman compression to the plain input text.
➢ Generate the cipher text in hexadecimal form (in the form of alphabets (A, B, C, D, E, F) and digits (0, 1, 2, 3, 4, 5, 6, 7, 8, 9)) by AES algorithm [13] which is applied to the compressed output.
➢ Take the first 7 characters of the hexadecimal values; this part will be hidden in the image.
➢ Take the rest of the hexadecimal values; this will form the key (**Key 3**).
➢ Hide the first 7 hex values in the Image as mentioned in 3.1.

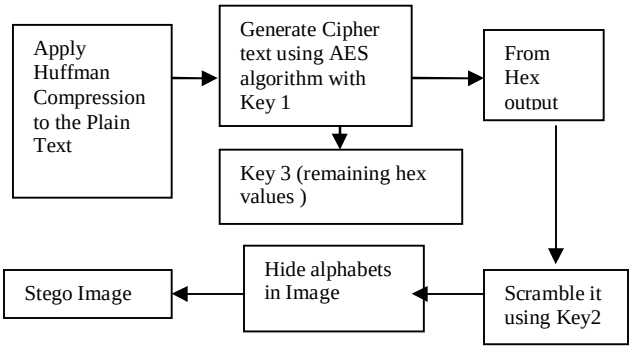


Figure9: Proposed System for hiding text

3.3.2 Retrieving Text

- Retrieve the 7 hexadecimal values from the image.
- Combine all the values with the help of Key3.
- Apply AES decryption technique to the retrieved hexadecimal values using Key1.
- Regenerate the original text message with the help of Huffman decompression algorithm.

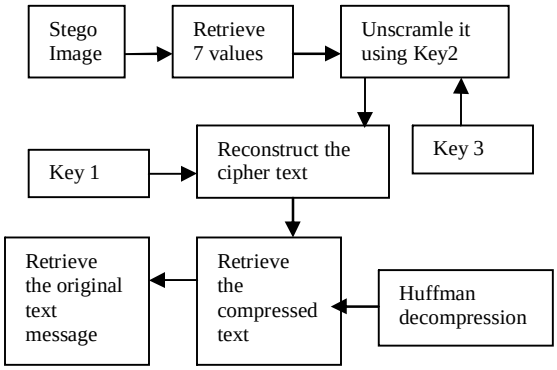


Figure10: Proposed System for retrieving text

5. How secure the proposed system is?

The proposed solution is highly secure since-

❖ **It's a combination of three techniques:**

a) Huffman Compression
b) AES for cryptography
c) DCT manipulation for Steganography.

❖ **Number of Keys:** This system contains total 3 keys.

a) One 128 bits private key for AES algorithm

b) One 56 bits private key for scrambling the cipher text.

c) One extra private generated key for retrieving the original message.

This extra generated key make the system highly secured.

5. Conclusion

The work accomplished during this paper can be summarized with the following points:

- In this paper we have presented a new system for the combination of **Compression, cryptography and Steganography using three keys** which could be proven a highly secured method for data communication in near future.
- Steganography especially combined with cryptography and compression is a powerful tool which enables people to communicate without possible eavesdroppers even knowing there is a form of communication in the first place. The proposed method provides acceptable image quality with very little distortion in the image.

5. References

[1] Domenico Daniele Bloisi , Luca Iocchi: Image based Steganography and cryptography, Computer Vision theory and applications volume 1 , pp. 127-134 .

[2] D.R. Stinson, Cryptography: Theory and Practice, Boca Raton, CRC Press, 1995. ISBN: 0849385210

[3] Mamta Sharma, S.L. Bawa D.A.V. college: Compression Using Huffman Coding, IJCSNS International Journal of Computer Science and Network Security, VOL.10 No.5, May 2010

[4] Kharrazi, M., Sencar, H. T., and Memon, N. (2004). Image Steganography: Concepts and practice. In WSPC Lecture Notes Series

[5] DAVID A. HUFFMAN+, ASSOCIATE, A Method for the Construction of Minimum-Redundancy Codes, PROCEEDINGS OF THE IRE.

[6] Daemen, Joan; Rijmen, Vincent. AES Proposal: Rijndael. Source: http://csrc.nist.gov/encryption/aes/rijndael/Rijndael.pdf

[7] Provos, N. and Honeyman, P. (2003). Hide and seek: An introduction to

steganography. IEEE SECURITY & PRIVACY
[8] Owens, M., "A discussion of covert channels and steganography", SANS Institute, 2002
[9] Chandramouli, R., Kharrazi, M. & Memon, N., "Image Steganography and steganalysis: Concepts and Practice", Proceedings of the 2nd International Workshop on Digital Watermarking, October 2003
[10] Jamil, T., "Steganography: The art of hiding information is plain sight", IEEE Potentials, 18:01, 1999
[11] Stefan Katznbeisser, Fabien.A., P.Petitcolas editors, Information Hiding Techniques for Steganography and Digital Watermarking, Artech House, Boston. London, 2000.
[12] Wang, H & Wang, S, "Cyber warfare: Steganography vs. Steganalysis", Communications of the ACM, 47:10, October 2004
[13] Dunbar, B., "Steganography techniques and their use in an Open-Systems environment", SANS Institute, January 2002
[14] C.E., Shannon, (1949), Communication theory of secrecy systems, Bell System Technical Journal, 28, 656-715.
[15] Marvel, L.M., Boncelet Jr., C.G. & Retter, C., "Spread Spectrum Steganography", IEEE Transactions on image processing, 8:08, 1999
[16] N. F. Johnson and S. Katzenbeisser, .A survey of steganographic techniques., in S. Katzenbeisser and F. Peticolas (Eds.): Information Hiding, pp.43-78. Artech House, Norwood, MA, 2000.
[17] Currie, D.L. & Irvine, C.E.,"Surmounting the effects of lossy compression on Steganography", 19th National Information Systems Security Conference, 1996
[18] G., Derrick, (2001), Data watermarking Steganography and watermarking of digital data, Computer Law & Security Report, 17 (2), 101-104.
[19] Ross J. Anderson, Fabien A.P. Petitcolas, "On The Limits of Steganography", IEEE Journal of Selected Areas in Communications, 16(4): 474-481, May 1998. Special Issue on Copyright & Privacy Protection. ISSN 0733-8716.
[20]http://www.codeproject.com/KB/library/ArisFFTDFTLibrary.aspx
[21]http://www.xdp.it